



Pemrograman Berorientasi Objek dengan C#

[Lecture Note Pertemuan ke - 11]
(Dynamic Link Library)

Ng Poi Wong, 2020, Sesi 5 : Dynamic Link Library, Lecture Notes, Pemrograman Berorientasi Objek dengan C# (IF0058), STMIK Mikroskil Medan, Dikirimkan 02 Maret 2020.

Capaian MK : Mahasiswa mampu memanfaatkan Dynamic Link Library

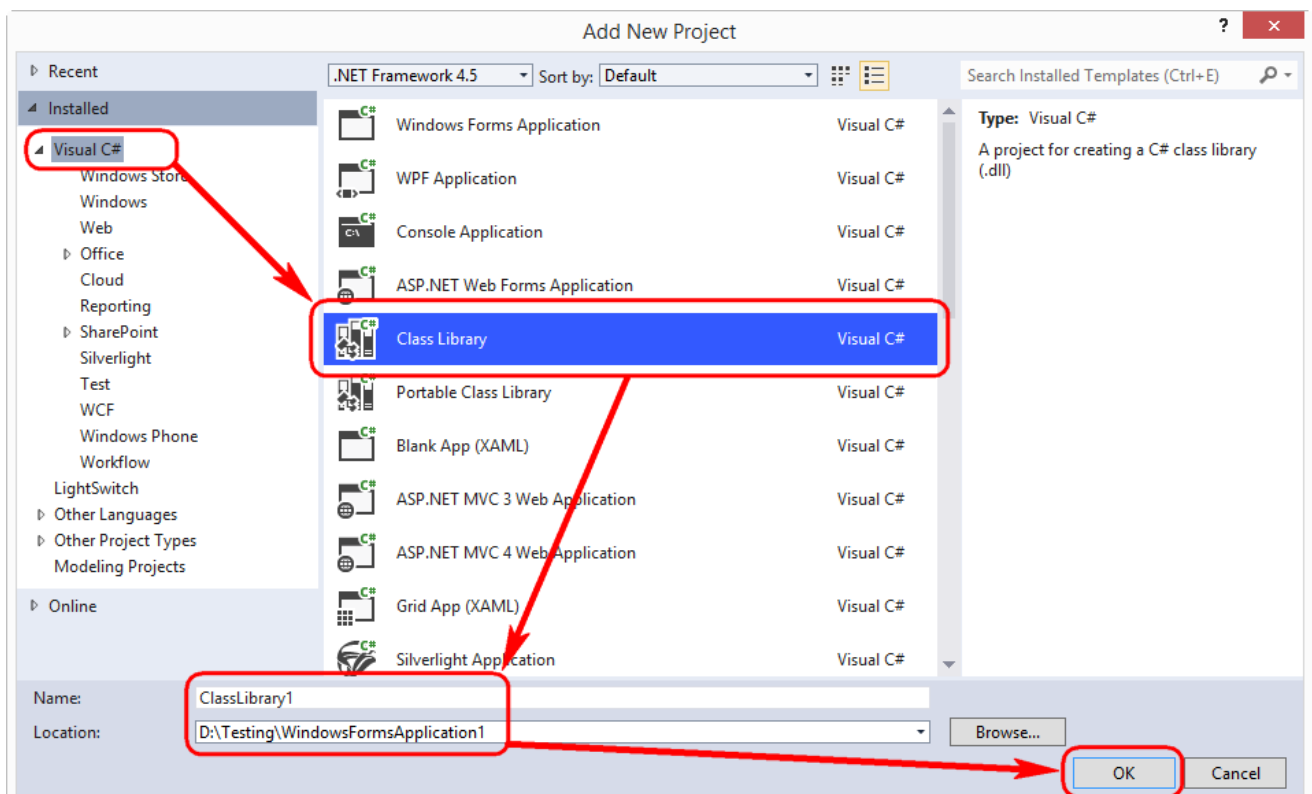
DAFTAR ISI PENJELASAN SLIDE

Slide 03 s/d 08	[Class Library]	2
Slide 09 s/d 16	[Form Dinamis]	4
Slide 17 s/d 24	[User Control]	5
Slide 25 s/d 27	[Windows Forms Control Library]	7

PENJELASAN DARI SLIDE ke-03 s/d 08

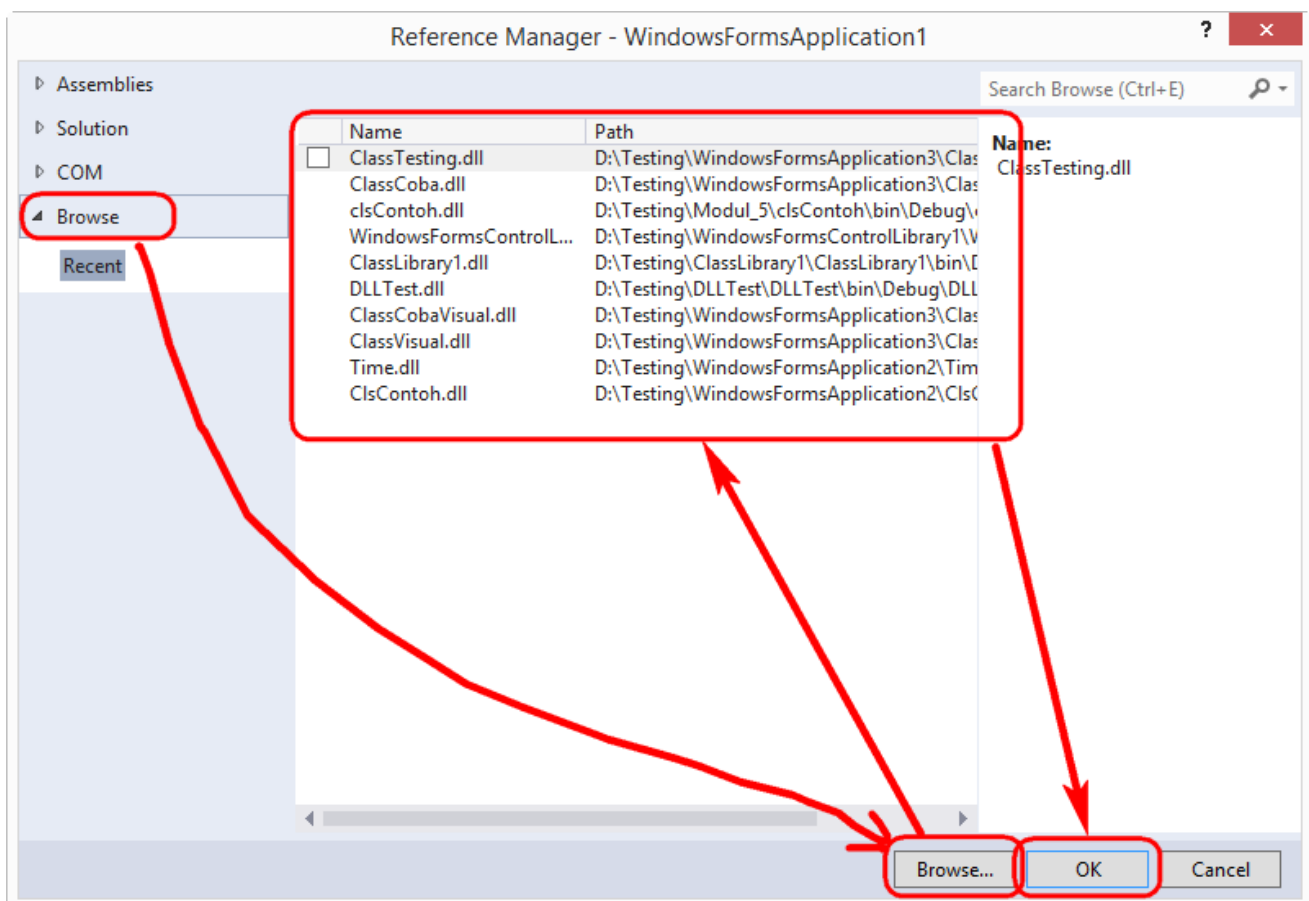
PENTING : Versi Visual Studio yang digunakan pada Lecture Note ini adalah Visual Studio 2012. Perbedaan versi Visual Studio dapat mempengaruhi lokasi letak project ataupun penamaan project.

- **Dynamic Link Library (DLL)** merupakan library buatan yang berisikan kumpulan fungsi dan prosedur yang dapat digunakan lintas project, lintas solution, bahkan lintas bahasa pemrograman.
- DLL pada Visual Studio terdiri dari DLL berbasis Teks (Console) dan DLL berbasis Visual (GUI).
- DLL berbasis Teks (Console) dibangun melalui **Class Library**.
- DLL berbasis Visual (GUI) dibangun melalui **Windows Forms Control Library**.
- Untuk penjelasan alasan diperlukan penggunaan **Class Library**, dapat di simak video Youtube berikut : https://youtu.be/btVT_5Xs06Q
- Untuk membangun **Class Library**, dapat dilakukan melalui :
 1. Menu **File | Add | New Project**, atau
 2. Klik kanan Solution, menu **Add | New Project**
 Maka akan muncul tampilan jendela **Add New Project**, Kemudian pilih pilihan project baru berupa **Class Library**, Sesuaikan nama project yang diinginkan, dan klik **OK**.



PENJELASAN DARI SLIDE ke-03 s/d 08

- **Class Library** yang telah selesai dibangun, wajib di-compile untuk dihasilkan file **DLL**.
- Untuk dapat memanfaatkan file **DLL** yang telah dibangun dengan **Class Library**, maka project yang akan memanfaatkan file **DLL** tersebut, wajib menambahkan referensi file **DLL** melalui :
 1. Menu **Project | Add Reference**, atau
 2. Klik kanan Project yang ingin memanfaatkan file **DLL**, menu **Add Reference**
 Maka akan muncul tampilan jendela **Reference Manager**,
 Kemudian pilih pilihan **Browse**,
 Kemudian klik tombol **Browse...** untuk memilih lokasi file DLL yang ingin digunakan,
 Setelah itu, maka akan muncul file DLL yang ingin digunakan di daftar DLL di tampilan bagian tengah
 (Pastikan file DLL yang terpilih telah tercentang di bagian depan nama file DLL).
 Setelah klik tombol **OK**.



- Untuk penjelasan cara membangun dan menggunakan **Class Library**, dapat di simak video Youtube berikut : <https://youtu.be/j0Sxn4kKzXA>

PENJELASAN DARI SLIDE ke-09 s/d 16

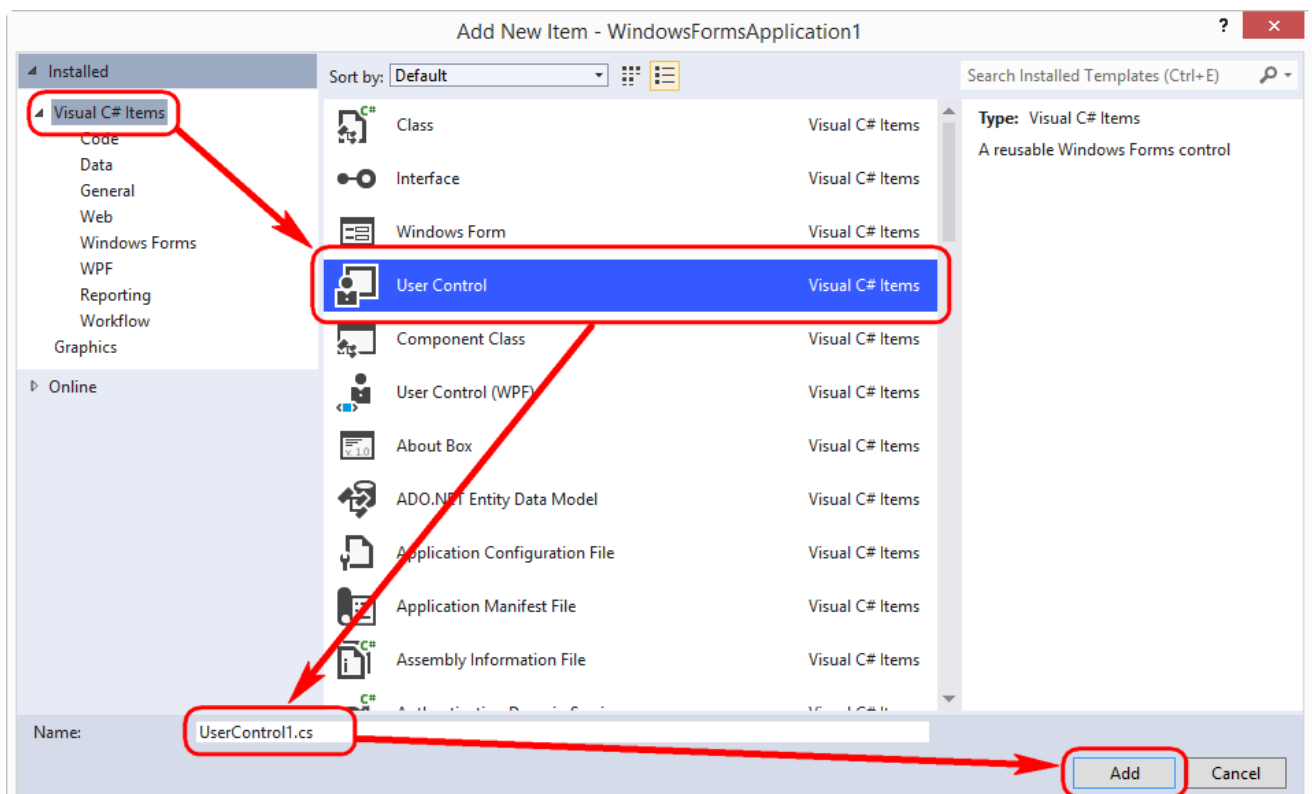
- Mulai dari **Sesi 3 tentang Form dan Event pada pertemuan ke-5** pada semester ini, telah dijelaskan materi tentang ruang lingkup kerja dari Form, dimana :
 1. Pada saat dipasangkan suatu objek kontrol ke dalam Form, maka secara auto akan di-generate sejumlah kode program ke dalam bagian **Form Designer** yang berfungsi untuk memasangkan suatu objek kontrol ke dalam Form.
 2. Pada saat diatur konfigurasi properties dari Form atau objek kontrol yang ada pada Form, maka secara auto akan di-generate sejumlah kode program ke dalam bagian **Form Designer** yang berfungsi untuk mengkonfigurasi properties dari Form atau objek kontrol yang ada pada Form.
 3. Pada saat dipanggil suatu event dari Form atau objek kontrol yang ada pada Form, maka secara auto akan di-generate sejumlah kode program ke dalam bagian **Form Designer** yang berfungsi untuk memanggil event dari Form atau objek kontrol yang ada pada Form.
- Semua kode program yang ter-generate secara auto ke dalam **Form Designer** adalah hasil konversi perilaku programmer berupa **Click & Drag** untuk merancang tampilan visual menjadi kode program.
- **Form Dinamis** adalah materi yang menjelaskan / mendemokan bagaimana kita merancang suatu Windows Forms Application dengan menggunakan kode program, dimana tidak akan menggunakan Click & Drag untuk merancang tampilan, sehingga tidak akan terjadi proses auto-generate kode program ke dalam Form Designer.

Istilah lainnya adalah memindahkan semua kode program yang bakal auto-generate dari Form Designer ke dalam tampilan View Code dari Form.
- Untuk penjelasan contoh sederhana membangun aplikasi dengan **Form Dinamis**, dapat di simak video Youtube berikut dengan beberapa skenario :
 1. Form Dinamis yang menggunakan objek kontrol dasar : <https://youtu.be/BKdRm9ExBcQ>
 2. Form Dinamis yang menggunakan objek kontrol kategori Container : <https://youtu.be/sq8gR7vU0S4>

PENJELASAN DARI SLIDE ke-17 s/d 24

PENTING : Versi Visual Studio yang digunakan pada Lecture Note ini adalah Visual Studio 2012. Perbedaan versi Visual Studio dapat mempengaruhi lokasi letak project ataupun penamaan project.

- **ToolBox** berisikan daftar objek kontrol yang umumnya digunakan pada Windows Forms Application, dimana objek kontrol tersebut merupakan objek kontrol built-in dari Visual Studio.
- **User Control** merupakan fitur untuk membangun objek kontrol buatan, dimana isi dari objek kontrol buatan tersebut tersusun dari objek kontrol built-in yang telah tersedia pada Toolbox.
- Objek kontrol buatan dengan **User Control**, akan muncul pada Toolbox dengan syarat : Project yang memiliki **User Control** tersebut wajib di-kompile minimal 1 (satu) kali.
- **User Control** yang telah dibangun pada suatu project, maka objek kontrol buatan tersebut hanya dapat digunakan pada project yang memiliki **User Control** tersebut.
- Untuk membuat **User Control**, dapat dilakukan melalui :
 1. Menu **Project | Add User Control**, atau
 2. Klik kanan Project yang ingin dibuatkan User Control, menu **Add | User Control**
 Maka akan muncul tampilan jendela **Add New Item**,
 Kemudian pilih pilihan item baru berupa **User Control**,
 Sesuaikan nama User Control yang diinginkan, dan klik **Add**.



PENJELASAN DARI SLIDE ke-17 s/d 24

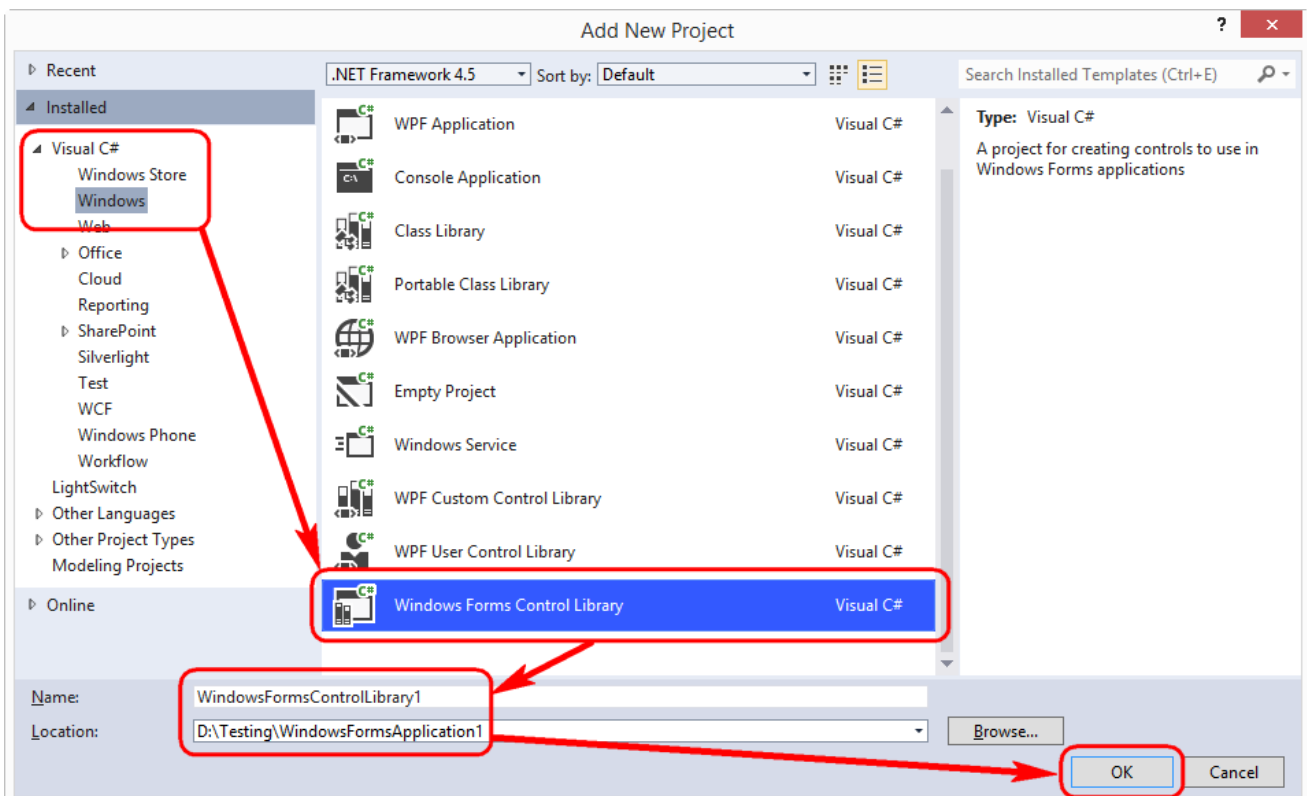
- Semua objek kontrol yang digunakan pada saat pembuatan **User Control**, hanya dikenal di dalam ruang lingkup dari pembuatan **User Control** tersebut.
- **User Control** yang telah menjadi ToolBox, tidak akan bisa lagi diakses nama, properties, maupun event dari objek kontrol yang ada di dalam **User Control** tersebut.
- Setiap objek kontrol dari hasil **User Control**, akan memiliki properties tersendiri.
- Properties dari objek kontrol dari hasil **User Control** dapat terdiri dari properties built-in dan properties buatan sendiri.
- Properties dari objek kontrol dari hasil **User Control** berfungsi sebagai media penghubung antara nama, properties, dan event dari objek kontrol yang ada di dalam **User Control** tersebut.
- Untuk penjelasan pembuatan dan penggunaan **User Control**, dapat di simak video Youtube berikut :
 1. Contoh User Control (Bagian 1) : <https://youtu.be/MtMmJmj2VyU>
 2. Contoh User Control (Bagian 2) : <https://youtu.be/MK Ib GglbA>

PENJELASAN DARI SLIDE ke-25 s/d 27

PENTING : - Versi Visual Studio yang digunakan pada Lecture Note ini adalah Visual Studio 2012. Perbedaan versi Visual Studio dapat mempengaruhi lokasi letak project ataupun penamaan project.

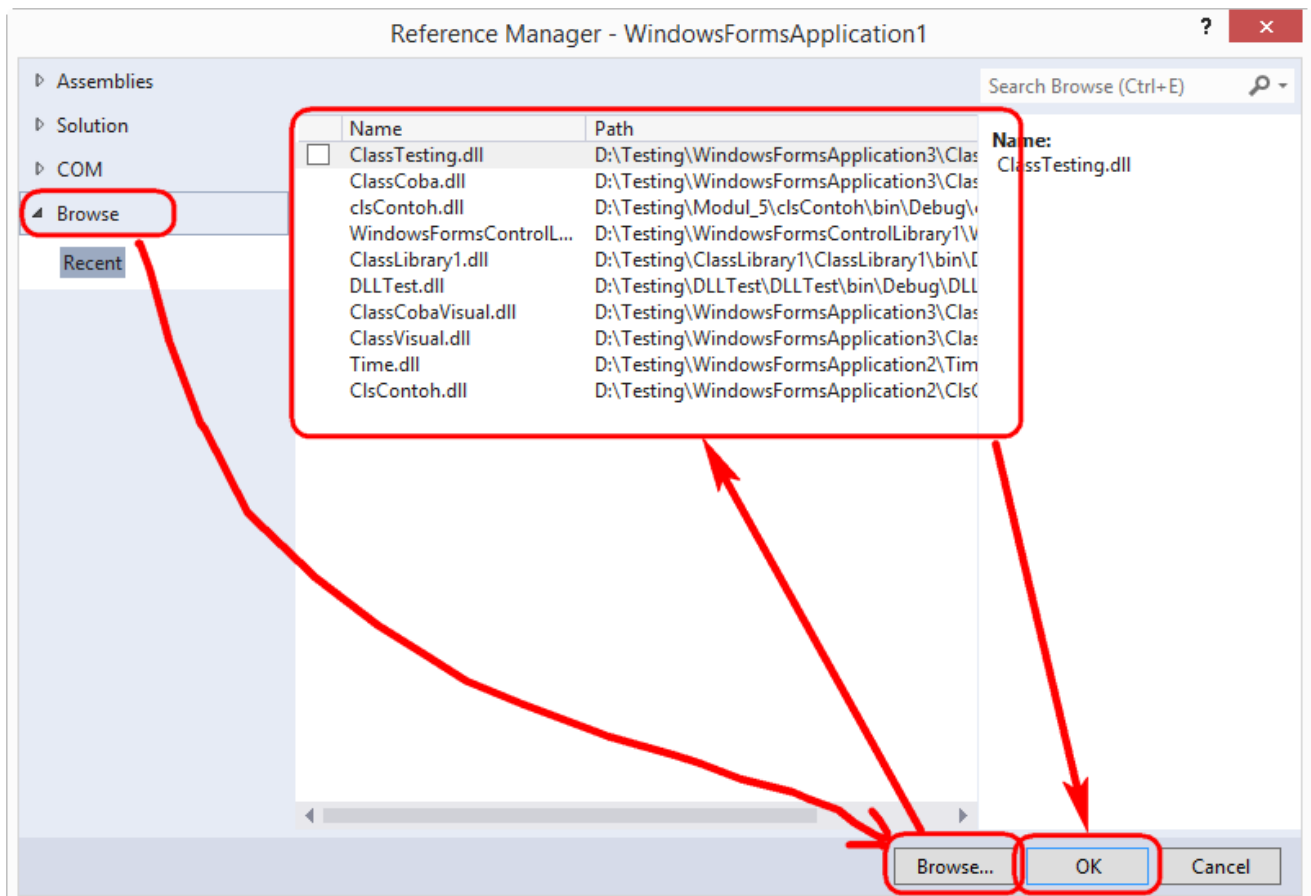
- **Wajib sudah memahami materi Class Library, Form Dinamis, dan User Control di atas.**

- **Windows Forms Control Library** merupakan DLL berbasis Visual (GUI), dimana di dalam Windows Forms Control Library berisikan kumpulan fungsi dan prosedur dalam bentuk **User Control** yang dapat digunakan lintas project, lintas solution, bahkan lintas bahasa pemrograman.
- Konsep ruang lingkup dari penerapan **Windows Forms Control Library** adalah sama dengan Class Library.
- Untuk membangun **Windows Forms Control Library**, dapat dilakukan melalui :
 1. Menu **File | Add | New Project**, atau
 2. Klik kanan Solution, menu **Add | New Project**
 Maka akan muncul tampilan jendela **Add New Project**,
 Kemudian pada menu sisi kiri, pilih kategori **Windows** dari bahasa yang digunakan.
 Kemudian pilih pilihan project baru berupa **Windows Forms Control Library**,
 Sesuaikan nama project yang diinginkan, dan klik **OK**.



PENJELASAN DARI SLIDE ke-25 s/d 27

- **Windows Forms Control Library** yang telah selesai dibangun, wajib di-compile untuk dihasilkan file **DLL**.
- Untuk dapat memanfaatkan file **Dynamic Link Library (DLL)** yang telah dibangun dengan **Windows Forms Control Library**, maka project yang akan memanfaatkan file **DLL** tersebut, wajib menambahkan referensi file **DLL** melalui :
 1. Menu **Project | Add Reference**, atau
 2. Klik kanan Project yang ingin memanfaatkan file **DLL**, menu **Add Reference**
 Maka akan muncul tampilan jendela **Reference Manager**,
 Kemudian pilih pilihan **Browse**,
 Kemudian klik tombol **Browse...** untuk memilih lokasi file DLL yang ingin digunakan,
 Setelah itu, maka akan muncul file DLL yang ingin digunakan di daftar DLL di tampilan bagian tengah (Pastikan file DLL yang terpilih telah tercentang di bagian depan nama file DLL).
 Setelah klik tombol **OK**.



- Untuk penjelasan cara membangun dan menggunakan **Windows Forms Control Library**, dapat di simak video Youtube berikut : <https://youtu.be/5upQ9oHJ21s>